(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 2 207 301 A1

(12) EUROPEAN PATENT APPLICATION
published in accordance with Art. 153(4) EPC

(43) Date of publication:
14.07.2010 Bulletin 2010/28

(21) Application number: **08783887.6**

(22) Date of filing: **06.08.2008**

(51) Int Cl.:
H04L 9/00 *(2006.01)*

(86) International application number:
PCT/CN2008/071894

(87) International publication number:
WO 2009/062415 (22.05.2009 Gazette 2009/21)

(84) Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MT NL NO PL PT RO SE SI SK TR
Designated Extension States:
AL BA MK RS

(30) Priority: **14.11.2007 CN 200710165986**

(71) Applicant: **Huawei Technologies Co., Ltd.**
Longgang District, Shenzhen
Guangdong 518129 (CN)

(72) Inventor: **SUN, Chao**
Guangdong 518129 (CN)

(74) Representative: **Charles, Glyndwr**
Reinhard, Skuhra, Weise & Partner GbR
Patent- und Rechtsanwälte
Friedrichstrasse 31
80801 München (DE)

(54) AN AUTHENTICATION METHOD FOR REQUEST MESSAGE AND THE APPARATUS THEREOF

(57) A method for authenticating request messages is disclosed. An authentication service device performs centralized allocation and management for authentication random numbers; when a User Equipment (UE) uses a protected service, the key negotiation process needs to be performed only once, whereupon the authentication is performed with multiple Application Servers (ASs) in turn according to the policy of using an authentication random number. Further, the corresponding authentication service device, AS, and UE are disclosed.

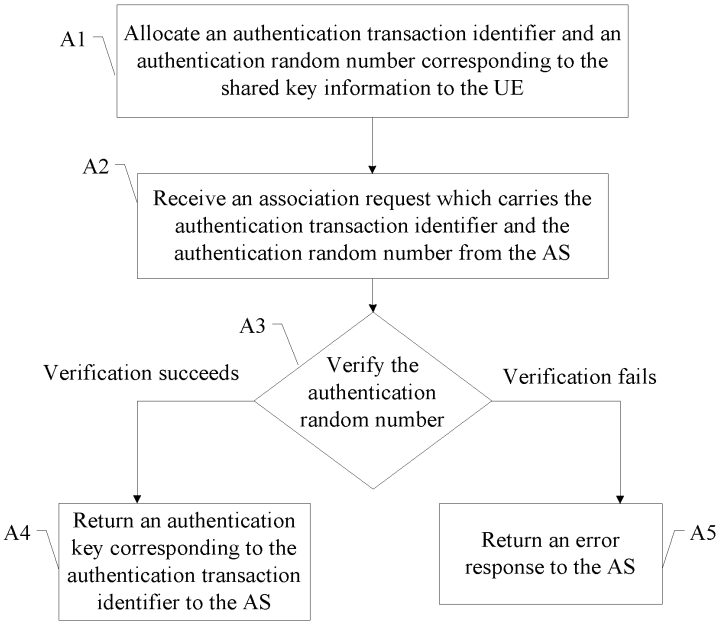

FIG. 2

EP 2 207 301 A1

Description

[0001] This application claims priority to Chinese Patent Application No. 200710165986.8, filed with the Chinese Patent Office on November 14, 2007 and entitled "Method for Authenticating Request Messages and Device Thereof', which is incorporated herein by reference in its entirety.

Field of the Invention

[0002] The present invention relates to communication technologies, and in particular, to a method for an authentication service device to authenticate request messages, a method for an Application Server (AS) to authenticate requests of a User Equipment (UE), a UE authentication method, and the corresponding authentication service device, AS, and UE.

Background of the Invention

[0003] With the diversification and development of communication services, both the service provider and the user require a reliable security mechanism to ensure legal use of services. For example, in the Internet Protocol Multimedia Subsystem (IMS)-based Multimedia Broadcast and Multicast Service (MBMS), the User Equipment (UE) is generally authenticated before the user accesses a protected MBMS service through the UE. FIG. 1 shows a common authentication process in the prior art. The authentication process includes two steps:

I. Key negotiation

[0004] Through the key negotiation process, the UE and an authentication service device that provides a security mechanism negotiate the shared key-related data. Specifically, the key negotiation is generally performed through a Generic Bootstrapping Architecture (GBA). In the GBA, the device that provides the security mechanism is a Bootstrapping Server Function (BSF). After completion of the GBA process, both the UE and the BSF have recorded the key-related data of the GBA process, including Bootstrapping Transaction Identifier (B-TID), IP Multimedia Private Identity (IMPI), Cipher Key (CK), and Integrity Key (IK). The UE and the BSF may use the CK and the IK to generate keys (Ks) shared between them.

II. Security association

[0005] The security association process associates the Ks negotiated between the UE and the BSF to the AS which provides the Network Application Function (NAF). After completion of the GBA process, the Ks is shared between the UE and the BSF, and the UE may use the AS identifier (NAF_Id) and the Ks to calculate the relevant keys (Ks_NAF). However, the AS has not obtained the key information yet. To negotiate the keys shared between the UE and the AS, the following steps need to be performed:

[0006] 1. The UE sends a HyperText Transfer Protocol (HTTP) request to the AS, and the request carries a B-TID.

[0007] 2. The AS sends an HTTP request to the BSF, and the request carries the B-TID and the host name of the AS.

[0008] 3. After the BSF authenticates the request of the AS, the BSF calculates out the relevant keys (Ks_NAF) according to the Ks negotiated with the UE and the received host name. Afterward, the BSF sends an HTTP response to the AS. The response may carry the validity period of the keys and the security settings of the user.

[0009] 4. After receiving the response from the BSF, the AS calculates the authentication response data according to the Ks_NAF, and sends the data to the UE through an HTTP response.

[0010] After the UE authenticates the response from the AS according to the recorded Ks_NAF, the authentication process is finished, and the UE and the AS may start secure communications through the established shared keys (Ks_NAF).

[0011] During the research and practice of the present invention, the inventor finds that to complete a sophisticated service, multiple service functions need to work together at the server side; namely, multiple ASs is needed to implement a complete service. For example, for the MBMS service on a mobile network, an AS1 which provides a demonstration portal enables the UE to obtain the service guide, and an AS2 which provides a subscription portal enables the user to subscribe to the MBMS service. In this case, according to the current authentication mode, the UE needs to perform the foregoing key negotiation process and security association process with different ASs. Too many message interactions are involved, and the application response speed is delayed.

Summary of the Invention

[0012] Embodiments of the present invention provide a request message authentication solution to simplify the authentication process between a UE and multiple applications.

[0013] A request message authentication method includes: negotiating key information with a UE, and allocating an authentication transaction identifier and an authentication random number corresponding to the key information to the UE; receiving an association request from an AS, where the association request carries the authentication transaction identifier and the authentication random number; verifying the authentication random number carried in the association request according to the authentication transaction identifier carried in the association request; and searching for the key information corresponding to the authentication transaction identifier carried in the association request after the verification succeeds, and returning an association response to the

AS in response to the association request, where the association response carries an authentication key generated according to the found key information.

[0014] A method for an AS to authenticate a request of a UE includes: receiving an authentication request from a UE, where the authentication request carries an authentication transaction identifier and an authentication random number; sending an association request to an authentication service device according to the authentication request, where the association request carries the authentication transaction identifier and the authentication random number; receiving an association response returned by the authentication service device in response to the association request, where the association response carries an authentication key; and returning a success response to the UE in response to the authentication request, where the success response carries the authentication response data generated according to the authentication key.

[0015] A UE authentication method includes: sending an authentication request to an AS, where the authentication request carries an authentication transaction identifier and an authentication random number corresponding to key information; receiving a success response returned by the AS in response to the authentication request, where the success response carries authentication response data; and verifying the authentication response data according to the key information.

[0016] An authentication service device includes: an authentication service module, adapted to: negotiate key information with a UE, and allocate an authentication transaction identifier and an authentication random number corresponding to the key information to the UE; and a uniform authentication module, adapted to: receive an association request from an AS, and obtain the authentication transaction identifier and the authentication random number carried in the association request; verify the authentication random number carried in the association request according to the authentication transaction identifier carried in the association request and the data provided by the authentication service module; and search for the key information corresponding to the authentication transaction identifier carried in the association request after the verification succeeds, and return an association response to the AS in response to the association request, where the association response carries an authentication key generated according to the found key information.

[0017] An AS includes: an interface module, adapted to: receive an authentication request from a UE; send an association request to an authentication service device; receive an association response returned by the authentication service device in response to the association request; and return a success response to the UE in response to the authentication request; and a security association module, adapted to: obtain an authentication transaction identifier and an authentication random number carried in the authentication request, and generate the association request according to the authentication request, where the association request needs to be sent to the authentication service device and carries the authentication transaction identifier and the authentication random number; obtain an authentication key carried in the association response, generate authentication response data according to the authentication key carried in the association response, and generate the success response that needs to be returned to the UE, where the success response carries the authentication response data.

[0018] A UE includes: a security service module, adapted to: negotiate key information with an authentication service device, and obtain an authentication transaction identifier and an authentication random number which are allocated by the authentication service device and correspond to the key information; an application service module, adapted to: send an authentication request to an AS, where the authentication request carries the authentication transaction identifier and the authentication random number; and obtain a success response returned by the AS in response to the authentication request, obtain the authentication response data carried in the success response, and verify the authentication response data according to the key information; and an authentication control module, adapted to: judge whether the key information and the authentication transaction identifier and the authentication random number corresponding to the key information are stored when an authentication request needs to be sent to the AS; if no such information is stored, control the security service module to obtain the required data; and, if such information is stored, control the application service module to send an authentication request.

[0019] In the technical solution under the present invention, the authentication service device performs centralized allocation and management for the authentication random numbers. When the UE uses a protected service, the key negotiation process needs to be performed only once, whereupon the authentication is performed with multiple ASs in turn according to the policy of using an authentication random number. Therefore, the process of authentication between the UE and the application is simplified, and the application response speed is improved.

Brief Description of the Drawings

[0020] FIG. 1 is a flowchart of an authentication method in the prior art;

[0021] FIG. 2 is a flowchart of an authentication method used by an authentication service device to authenticate a request message in an embodiment of the present invention;

[0022] FIG. 3 shows a logical structure of an authentication service device in an embodiment of the present invention;

[0023] FIG. 4 shows an application system architec-

ture of an authentication service in an embodiment of the present invention;

[0024] FIG. 5 is a flowchart of an authentication method used by an AS to authenticate a request of a UE in an embodiment of the present invention;

[0025] FIG. 6 shows a logical structure of an AS in an embodiment of the present invention;

[0026] FIG. 7 is a flowchart of a UE authentication method in an embodiment of the present invention;

[0027] FIG. 8 shows a logical structure of a UE in an embodiment of the present invention;

[0028] FIG. 9 shows a GBA process in an embodiment of the present invention;

[0029] FIG. 10 shows a successful authentication process in an embodiment of the present invention;

[0030] FIG. 11 shows an unsuccessful authentication process in an embodiment of the present invention; and

[0031] FIG. 12 shows another unsuccessful authentication process in an embodiment of the present invention.

Detailed Description of the Invention

[0032] A request message authentication solution is provided in an embodiment of the present invention. The solution is detailed below in different point of view with respect to an authentication service device, an AS, and a UE.

[0033] FIG. 2 is a flowchart of an authentication method used by an authentication service device to authenticate a request message in an embodiment of the present invention. The authentication method includes the following steps:

[0034] A1. An authentication service device negotiates key information with a UE, and allocates an authentication transaction identifier and an authentication random number corresponding to the key information to the UE.

[0035] This embodiment does not limit the mode of negotiation of key between the authentication service device and the UE. Any key negotiation mode is appropriate only if it enables both parties to obtain shared key information through the negotiation. The negotiation mode in use needs to be extended so that the authentication random number allocated by the authentication service device can be delivered to the UE.

[0036] For example, if the key negotiation is done through a GBA process. The Ks may be regarded as shared key information, and the B-TID allocated in the GBA process may be used as an authentication transaction identifier. The authentication random number may be carried in a new parameter (such as nextnonce) which is obtained by extending the response message to the UE.

[0037] The authentication service device may manage the authentication random number in two management modes:

[0038] (1) Update management

[0039] The authentication service device may allocate a new authentication random number to UE in response to each subsequent authentication request of the UE. For example, after completion of the key negotiation, the authentication service device sends a success response through which the first authentication random number is allocated to the UE. The first authentication random number is used when the UE sends an authentication request to the AS for the first time. Subsequently, whenever responding to the association request of the AS successfully, the authentication service device returns a newly allocated authentication random number which is delivered to the UE through the AS and is used in the next authentication request of the UE.

[0040] (2) Use Count management

[0041] The authentication service device may record the use count of the allocated authentication random number. For example, after completion of the key negotiation, the authentication service device sends a success response through which the first authentication random number is allocated to the UE, and the use count of the authentication random number is initialized to 1; subsequently, whenever responding to the association request of the AS successfully, the authentication service device increases the use count of the authentication random number progressively. In this mode, the UE also needs to record the use count of the authentication random number and increase the use count accordingly. The authentication request sent to the AS carries the use count of the authentication random number. The use count is further carried in the association request of the AS to the authentication service device, and is used as a basis for the authentication service device to verify the request message.

[0042] A2. The authentication service device receives the association request from the AS, where the association request carries the authentication transaction identifier and the authentication random number.

[0043] If the management mode (2) is applied, the association request received by the authentication service device further carries the use count of the authentication random number.

[0044] A3. The authentication service device verifies the authentication random number carried in the association request according to the authentication transaction identifier carried in the received association request. If the verification succeeds, the process proceeds to step A4; and, if the verification fails, the process goes to step A5.

[0045] Based on the management mode (1), the authentication service device verifies the authentication random number by: searching for the authentication random number corresponding to the authentication transaction identifier carried in the association request, and verifying whether the authentication random number carried in the association request is consistent with the found authentication random number.

[0046] Based on the management mode (2), the authentication service device verifies the authentication

random number by: searching for the authentication random number and the use count of the authentication random number corresponding to the authentication transaction identifier carried in the association request, and verifying whether the authentication random number and its use count carried in the association request are consistent with those found authentication random number and its use count.

[0047] A4. After the verification succeeds, the authentication service device searches for the key information corresponding to the authentication transaction identifier carried in the association request, and returns an association response to the AS in response to the association request, where the association response carries an authentication key generated according to the found key information.

[0048] For example, if the MBMS Request Key (MRK) is negotiatied through a GBA process, the authentication service device may search for the corresponding Ks according to the B-TID carried in the association request, calculate the MRK according to the Ks, and return the MRK to the AS as an authentication key.

[0049] Based on the management mode (1), after the verification succeeds, the authentication service device needs to allocate a new authentication random number to update the found authentication random number. The returned association response carries the new authentication random number.

[0050] Based on the management mode (2), after the verification succeeds, the authentication service device needs to increase the use count of the found authentication random number progressively.

[0051] A5. After the verification fails, the authentication service device returns an error response to the AS in response to the association request.

[0052] The verification fails if the information to be verified is inconsistent with the found information (for example, if the authentication random number is inconsistent with the stored authentication random number, or if the use count of the authentication random number is inconsistent with the stored use count), or if the relevant information is not retrievable according to the authentication transaction identifier.

[0053] After the verification fails, the authentication service device may return an association failure response indicative of verification failure, and, through the AS, instruct the UE to perform the key negotiation process again. To further shorten the process, the authentication service device may deal with different failure circumstances accordingly, and return different error responses corresponding to different circumstances. For example, the authentication service device may further check for existence of any key information corresponding to the authentication transaction identifier carried in the association request. If such key information exists, it indicates that the key information shared with the UE still exists. The authentication service device may further check the validity of the key information. Therefore, the authentication service device may reallocate an authentication random number and use it as an authentication random number corresponding to the authentication transaction identifier carried in the association request, and return an association reset response to the AS in response to the association request. The association reset response carries the reallocated authentication random number, and may further carry an error cause. In this way, the UE may use the reallocated authentication random number to resend an authentication request to the AS. If no key information corresponding to the authentication transaction identifier carried in the association request exists, the authentication service device returns an association failure response to the AS and requires the UE to perform the key negotiation process again. The association failure response corresponds to the association request and indicates verification failure, and may further carry a cause of error.

[0054] The foregoing detailed error response mode is applicable to both the management mode (1) and the management mode (2). In the management mode (2), the authentication service device and the UE need to not only update the stored authentication random number, but also initialize the use count of the authentication random number.

[0055] Given below is an embodiment of the authentication service device which executes the foregoing method for authenticating request messages. To minimize the change to the existing device, the authentication service device in this embodiment inherits the existing modules of authentication service functions and adds a uniform authentication module which provides extended functions. As shown in FIG. 3, the authentication service device in this embodiment includes: an authentication service module 101 and a uniform authentication module 102.

[0056] The authentication service module 101 is adapted to: negotiate key information with a UE, and allocate an authentication transaction identifier and an authentication random number corresponding to the key information to the UE. This module performs the general authentication service functions, and the extended function is to deliver the authentication random number to the UE.

[0057] The uniform authentication module 102 is adapted to: receive an association request from an AS, and obtain the authentication transaction identifier and the authentication random number carried in the association request; verify the authentication random number carried in the association request according to the authentication transaction identifier carried in the association request and the data provided by the authentication service module 101; and search for the key information corresponding to the authentication transaction identifier carried in the association request after the verification succeeds, and return an association response to the AS in response to the association request, where the association response carries an authentication key generated according to the found key information.

[0058] To reduce the load of information query between modules, the uniform authentication module 102 is further adapted to store the data found in the authentication service module 101 for verifying the authentication random number carried in the association request. In this way, when the uniform authentication module 102 verifies the authentication random number carried in the association request, the uniform authentication module 102 may query the stored data first, and, if the query fails, query the authentication service module 101. After receiving a query request from the uniform authentication module 102, the authentication service module 101 may return a response which carries the corresponding authentication random number, key information etc altogether. In this way, the uniform authentication module 102 may search the stored data for the key information directly after the verification succeeds.

[0059] Based on different modes of managing the authentication random number, the uniform authentication module 102 verifies the authentication random number carried in the association request by: verifying whether the authentication random number carried in the association request is consistent with the stored authentication random number, or verifying whether the authentication random number carried in the association request and its use count are consistent with the stored authentication random number and its use count (in the latter case, the uniform authentication module 102 further obtains the use count of the authentication random number carried in the association request). In the former case, the uniform authentication module 102 is further adapted to: allocate a new authentication random number to update the stored authentication random number after the verification succeeds, and add the new authentication random number into a returned association response for instructing the UE to use the new authentication random number to send an authentication request to the next AS. In the latter case, the uniform authentication module 102 is further adapted to increase the stored use count of the authentication random number progressively after the verification succeeds.

[0060] To deal with the process of verification failure, the uniform authentication module 102 is further adapted to return an association failure response to the AS in response to the received association request. The association failure response indicates that the verification fails and the UE needs to perform key negotiation again. The uniform authentication module 102 may be further adapted to execute the foregoing detailed error response solution. That is, after the verification fails, the uniform authentication module 102 checks for existence of any key information corresponding to the authentication transaction identifier carried in the association request. If such key information exists, the uniform authentication module 102 reallocates an authentication random number to update the stored authentication random number corresponding to the authentication transaction identifier carried in the association request, and returns an association reset response to the AS in response to the association request. The association reset response carries the reallocated authentication random number for instructing the UE to use the reallocated authentication random number to resend an authentication request to the AS. If no such key information exists, with reference to the foregoing processing of verification failure, the uniform authentication module 102 returns an association failure response to the AS in response to the association request.

[0061] In a Generic Authentication Architecture (GAA), the authentication service module in this embodiment can be implemented with reference to the BSF. As a new module, the uniform authentication module, also named as the Authentication Server (AuthS) may be integrated with the BSF or implemented as an authentication server independent from the BSF. FIG. 4 shows an application system architecture of an AuthS in the case that the AuthS is integrated with the BSF. A GAA-based Ub interface exists between the BSF and the UE, and a GAA-based Ua interface exists between the UE and each AS. The format of the interface message between the AuthS and each AS is similar to the format of the message between the UE and the AS, or the XML format. The specific bearer protocol (such as HTTP and HTTPS) is not limited herein. The interface between the AuthS and the BSF is an internal interface, and is not necessarily defined. If the AuthS is independent, the interface between the AuthS and the BSF is similar to the Zn interface between the AS and the BSF in the GAA, which can be used as a reference. To make the illustration clearer, FIG. 4 does not show the connection between the AS and the BSF. To be compatible with the existing GAA authentication mode, the AS may maintain the connection with the BSF. In the service implementation process, the existing authentication mode (using the interface to the BSF) or the authentication mode provided herein (using the interface to the AuthS) is selected by judgment according to the authentication request of the UE.

[0062] FIG. 5 shows an authentication method used by an AS to authenticate a request of a UE in an embodiment of the present invention. The authentication method in this embodiment corresponds to the authentication method used by an authentication service device to authenticate a request message in the preceding embodiment. The authentication method in this embodiment includes the following steps:

[0063] B1. The AS receives an authentication request from a UE, where the authentication request carries an authentication transaction identifier and an authentication random number.

[0064] If the management mode (2) is applied, this authentication request further carries the use count of the authentication random number.

[0065] To be compatible with the existing authentication mode, the AS may select an adaptable authentication mode first according to the specific conditions of the authentication request after receiving the authentication

request from the UE. For example, in the current GAA process, the authentication request sent by the UE further carries a domain name of the server which authenticates the request. The domain name is usually placed in the last part of the realm parameter, and is delimited from the first half part by an @ symbol. After receiving the request, the AS may check whether the domain name behind the @ symbol of the realm parameter is consistent with the domain name of the AS. If they are consistent, the AS applies the existing GAA authentication process; if they are not consistent, and the domain name is the domain name of the authentication service device (for example, the domain name of the BSF), the AS executes the authentication process provided in this embodiment.

[0066] B2. The AS sends an association request to the authentication service device according to the received authentication request, where the association request carries the authentication transaction identifier and the authentication random number in the authentication request.

[0067] If the management mode (2) is applied, this association request further carries the use count of the authentication random number.

[0068] B3. The AS performs the corresponding subsequent operations according to different response results of the authentication service device, including:

[0069] B31. In the case of request success: The AS receives the association response returned by the authentication service device in response to the sent association request, where the association response carries an authentication key. The AS records the authentication key, and calculates the authentication response data according to the authentication key. Afterward, the AS returns a success response to the UE in response to the authentication request sent by the UE. The success response carries the generated authentication response data. Further, the AS may use the authentication key to authenticate the authentication request of the UE first, and return a success response to the UE after the authentication succeeds. If the management mode (1) is applied, the received association response further carries a new authentication random number. Therefore, the success response returned to the UE needs to further carry the new authentication random number for instructing the UE to use this new authentication random number to send an authentication request to the next AS.

[0070] B32. In the case of request failure: The AS receives an association failure response returned by the authentication service device in response to the sent association request, and returns a failure response to the UE in response to the authentication request sent by the UE according to the association failure response. The failure response carries an indication which requires the UE to perform key negotiation again.

[0071] B33. In the case of request resetting: The AS receives an association reset response returned by the authentication service device in response to the sent association request, where the association reset response carries the reallocated authentication random number. The AS returns a reset response to the UE in response to the authentication request sent by the UE. The reset response carries the reallocated authentication random number for instructing the UE to use this reallocated authentication random number to resend an authentication request.

[0072] Given below is an embodiment of the AS which executes the foregoing method for authenticating a request of a UE. As shown in FIG. 6, the AS includes: an interface module 201 and a security association module 202.

[0073] The interface module 201 is adapted to: receive an authentication request from a UE; send an association request to an authentication service device; receive an association response returned by the authentication service device in response to the sent association request; and return a success response to the UE in response to the authentication request sent by the UE.

[0074] If an error response is returned by the authentication service device, the interface module 201 is further adapted to: receive an association failure response or an association reset response returned by the authentication service device in response to the sent association request; and return a failure response or a reset response to the UE in response to the authentication request sent by the UE.

[0075] The security association module 202 is adapted to: obtain an authentication transaction identifier and an authentication random number carried in an authentication request which is received by the interface module 201, generate an association request which needs to be sent to the authentication service device, where the association request carries the authentication transaction identifier and the authentication random number in the authentication request; obtain an authentication key carried in an association response which is received by the interface module 201, calculate the authentication response data according to the authentication key, and generate a success response that needs to be returned to the UE, where the success response carries the authentication response data. Further, the security association module 202 may use the authentication key to authenticate the authentication request of the UE first, and generate a success response after the authentication succeeds.

[0076] If an error response is returned by the authentication service device, the security association module 202 is further adapted to: generate a failure response that needs to be returned to the UE according to the association failure response received by the interface module 201, where the failure response carries an indication which instructs the UE to perform key negotiation again; obtain the reallocated authentication random number carried in the association reset response according to the association reset response received by the interface module 201; and generate a reset response that needs to be returned to the UE, where the reset response carries

the reallocated authentication random number, with a view to instructing the UE to use the reallocated authentication random number to resend an authentication request.

[0077] To be compatible with the existing authentication mode, the security association module 202 is further adapted to obtain the domain name of the server which authenticates the request, where the domain name is carried in the authentication request. In this case, the security association module 202 may be further adapted to: check that the domain name (carried in the authentication request) of the server which authenticates the authentication request is the domain name of the authentication service device, and then generate an association request that needs to be sent to the authentication service device; and perform the subsequent operations according to the existing authentication mode if determining that the domain name (carried in the authentication request) of the server which authenticates the authentication request is its own domain name.

[0078] FIG. 7 is a flowchart of a UE authentication method in an embodiment of the present invention. The authentication method in this embodiment corresponds to the authentication method used by an authentication service device to authenticate a request message and the authentication method used by an AS to authenticate a request of a UE in the preceding embodiments. The authentication method in this embodiment includes the following steps:

[0079] C1. The UE judges whether any key information, and the authentication transaction identifier and the authentication random number corresponding to the key information are stored. If no such information is stored, the process proceeds to step C2; if such information is stored, the process goes to step C3.

[0080] C2. The UE negotiate a key information with the authentication service device, obtaining an authentication transaction identifier and an authentication random number which are allocated by the authentication service device and correspond to the key information. The process proceeds to step C3.

[0081] C3. The UE sends an authentication request to an AS, where the authentication request carries the stored authentication transaction identifier and authentication random number.

[0082] If the management mode (2) is applied, this authentication request further carries the use count of the authentication random number.

[0083] C4. The UE performs the corresponding subsequent operations according to different response results of the AS, including:

[0084] C41. In the case of request success: The UE receives a success response returned by the AS in response to the sent authentication request, where the success response carries authentication response data; and verifies the authentication response data according to the stored key information. If the management mode (1) is applied, the received success response further carries a new authentication random number, and therefore, the UE uses the new authentication random number to update the stored authentication random number; if the management mode (2) is applied, the UE increases the stored use count of the authentication random number progressively.

[0085] C42. In the case of request failure: The UE receives a failure response returned by the AS in response to the sent authentication request, where the failure response carries an indication which instructs the UE to perform key negotiation again. According to the indication, the UE clears the stored key information and the corresponding authentication transaction identifier and authentication random number, negotiate the key information with the authentication service device again, and then uses the re-negotiated key information to send an authentication request to the AS, namely, starts from step C2.

[0086] C43. In the case of request resetting: The UE receives a reset response returned by the AS in response to the sent authentication request. The reset response carries the reallocated authentication random number. The UE uses the reallocated authentication random number to update the stored authentication random number, and then uses the reallocated authentication random number to send an authentication request to the AS, namely, starts from step C3. If the management mode (2) is applied, the UE needs to initialize the use count of the authentication random number when updating the stored authentication random number.

[0087] Given below is an embodiment of the UE which executes the foregoing authentication method. As shown in FIG. 8, the UE includes:

a security service module 301, adapted to: negotiate key information with an authentication service device, and obtain an authentication transaction identifier and an authentication random number which are allocated by the authentication service device and correspond to the key information;

an application service module 302, adapted to: send an authentication request to the AS, where the authentication request carries the authentication transaction identifier and the authentication random number obtained by the security service module 301; and obtain a success response returned by the AS in response to the sent authentication request, obtain the authentication response data carried in the success response, and verify the authentication response data according to the key information obtained by the security service module 301; and

an authentication control module 303, adapted to: judge whether the key information and the authentication transaction identifier and the authentication random number corresponding to the key information are stored when an authentication request needs to be sent to the AS; if no such information is stored, control the security service module 301 to

obtain the required data; and, if such information is stored, control the application service module 302 to send an authentication request.

[0088] Further, if the management mode (1) is applied, the application service module 302 is further adapted to obtain a new authentication random number from the success response. In this case, the authentication control module 303 is further adapted to use the new authentication random number to update the stored authentication random number.

[0089] Further, if the management mode (2) is applied, the application service module 302 is further adapted to add the use count of the authentication random number into the authentication request. In this case, the authentication control module 303 is further adapted to increase the stored use count of the authentication random number progressively after the application service module 302 obtains the success response.

[0090] In the case of request failure, the application service module 302 is further adapted to: obtain a failure response or a reset response returned by the AS in response to the sent authentication request, obtain the indication in the failure response which instructs the UE to perform key negotiation again, and obtain the reallocated authentication random number in the reset response. The authentication control module 303 is further adapted to: clear the stored key information and the corresponding authentication transaction identifier and authentication random number according to the indication in the failure response, control the security service module 301 to obtain the required data again, and then control the application service module 302 to resend an authentication request; or use the reallocated authentication random number to update the stored authentication random number according to the indication in the reset response, and then control the application service module 302 to resend an authentication request.

[0091] In the foregoing embodiment, the authentication service device performs centralized allocation and management for the authentication random number. When the UE uses a protected service, the key negotiation process needs to be performed only once, whereupon the authentication is performed with multiple ASs in turn according to the policy of using the authentication random number. Therefore, the process of authentication between the UE and the application is simplified, and the application response speed is improved. Further, the AS may analyze the authentication request of the UE, and choose whether to execute the existing authentication mode or the authentication mode provided herein, thus enhancing the compatibility of the solution under the present invention.

[0092] To help understand the foregoing embodiments, application instances of the foregoing embodiments are given below, supposing that: the GAA serves as a basis; the system architecture shown in FIG. 4 is applied; and the AuthS manages the authentication random number in mode (2).

I. GBA process

[0093] When using a protected service, the UE accesses the BSF first and performs the GBA process which is extended according to the method provided in an embodiment of the present invention, then negotiates the key data shared with the AS. As shown in FIG. 9, the process includes the following steps:

[0094] 1. The UE sends an HTTP request to the BSF to perform a bootstrapping process. The request message includes the user ID information such as an International Mobile Subscriber Identification Number (IMSI) identifiable to the BSF. The UE generally lets the realm parameter of the Authorization header of the request message carry the Fully Qualified Domain Name (FQDN) of the BSF expected to be accessed.

[0095] 2. The BSF obtains the security information (such as an Authentication Vector (AV)) of the UE from the Home Subscriber Server (HSS) through a reference point "Zh". The AV includes the data such as CK and IK. The signaling process is detailed in 3GPP TS 29.109.

[0096] 3. The BSF selects an AV, calculates the authentication data, and sends an HTTP 401 Unauthorized response which carries the authentication data to the UE. The process of constructing a response is detailed in 3GPP TS 24.109.

[0097] 4. The UE checks the response of the BSF according to the stored data (such as data in the SIM). If the authentication check succeeds, the network is authenticated successfully. The UE calculates out the CK and IK. In this way, both the BSF and the UE own the IK and the CK.

[0098] 5. The UE calculates out the data in the Authorization header, and resends an HTTP request to the BSF.

[0099] 6. The BSF authenticates the request message of the UE. After the authentication succeeds, the BSF records the user ID, generates and stores the key data (including B-TID, IMPI, CK, and IK) in this GBA process, and generates key information Ks through the CK and the IK. The process of generating the key data is detailed in 3GPP TS 33.220.

[0100] 7. The BSF sends a 200 OK response to the UE, indicating authentication success. The response carries the lifespan of the B-TID and the Ks, and a nextnonce parameter added through extension. This parameter carries an authentication random number.

[0101] 8. The UE receives the response, and stores the parameters such as B-TID and nextnonce after the authentication succeeds, and generates the Ks according to the CK and the IK.

II. Authentication success process

[0102] It is assumed that the UE accesses two application services (AS1 and AS2) in turn according to the service process, as illustrated in FIG. 10 (because the

AuthS and the BSF work together as an authentication service device, FIG. 10 uses dotted lines to indicate the AuthS and the BSF and the relations between them, which also applies hereinafter). The process includes the following steps:

[0103] 1. The UE accesses the protected service, and the AS1 requires to authenticate the request message. The UE checks for existence of any key data generated in the GBA process. If any such key data exists, the UE calculates the MRK 1 for request authentication. The calculation method is described in 3GPP TS 33.220 Annex B. Instead of using the identifier of the AS1, the NAF_Id may be composed of the FQDN of the BSF and the security protocol identifier of the Ua interface. If no key data exists, the UE needs to execute the foregoing GBA process first.

[0104] 2. The UE sends an HTTP request carrying an Authorization header to the AS1. The Authorization header defines the parameters required for the AS1 to verify the message, including username, realm, nonce, and nc. The username parameter indicates the B-TID allocated by the BSF in the GBA process; the realm parameter is divided into two parts by a @ symbol, and the last part indicates the domain name of the server which authenticates the message, namely, the FQDN of the BSF; the nonce parameter is the nextnonce parameter value carried in the response returned by the BSF in the GBA process; and the nc parameter indicates the use count of the nextnonce. Other parameters are described in 3GPP TS 24.109.

[0105] 3. After receiving the request from the UE, the AS1 checks whether the domain name is consistent with the domain name of the AS1 according to the parameters in the Authorization header. If the domain name is consistent, the message authentication process defined in the standard is applied; otherwise, if the domain name is the domain name of the BSF, the authentication process provided herein is applied. Supposing that the domain name is the domain name of the BSF, the AS1 sends a request for the MRK to the AuthS. The request carries the authentication-related parameters such as B-TID, nonce, and nc.

[0106] 4. After receiving the request message, the AuthS retrieves the data stored in the AuthS according to the B-TID. If the retrieval fails, and if determining that it is the first authentication request of the UE according to nc = 1 in the request message, the AuthS sends a request to the BSF to obtain the data such as Ks negotiated in the GBA process.

[0107] 5. The BSF responds to the query request of the AuthS. The response carries the data such as Ks, Nonce, and data validity period corresponding to the B-TID. The AuthS stores the relevant data entries and initializes "nc = 1" in the entries.

[0108] 6. The AuthS checks whether the nonce and nc in the request message of the AS1 are consistent with the data stored in the AuthS itself. If the check succeeds, the AuthS generates the MRK in the same way as the UE, sends the MRK to the AS1 as a response, and increases the nc value in the entries progressively. If the management mode (1) is applied, the AuthS may further return a nextnonce parameter which carries the nonce for use in the next authentication.

[0109] 7. After receiving the response from the AuthS, the AS1 generates a response message to the UE, and calculates the authentication response data in the response message according to the MRK.

[0110] 8. The UE verifies the authentication response data of the AS1. After the verification succeeds, the UE increases the stored nc value progressively, thus indicating that the nonce has been applied once for authenticating the request message. Afterward, depending on the requirement of the service process, the UE accesses the second application service, namely, the AS2.

[0111] 9. The UE sends an HTTP request carrying an Authorization header to the AS2. This step differs from step 2 in that the nc value is an updated value.

[0112] 10. With reference to step 3, after receiving the request message of the UE, the AS2 sends a request for the MRK to the AuthS. The request carries the authentication-related parameters such as B-TID, nonce, and nc.

[0113] 11. After receiving the request, the AuthS retrieves the data stored in the AuthS according to the B-TID. The retrieval succeeds because the data related to the B-TID is already stored. The AuthS verifies whether the nonce and nc in the request are consistent with the stored values. If they are consistent, the AuthS responds to the MRK request of the AS2, and increases the nc value in the entries progressively.

[0114] 12. With reference to step 7, after receiving the response from the AuthS, the AS2 generates a response message to the UE, and calculates the authentication response data in the response message according to the MRK. The UE verifies the authentication response data of the AS2. After the verification succeeds, the UE increases the stored nc value progressively, thus indicating that the nonce has been applied once more for authenticating the request message.

III. Authentication failure process (1)

[0115] As shown in FIG. 11, the process includes the following steps:

[0116] 1-3. Steps 1-3 are the same as steps 1-3 in "II. Authentication success process".

[0117] 4. After receiving the request message, the AuthS retrieves the data stored in the AuthS according to the B-TID. If the retrieval fails and the nc is greater than 1(nc>1), or if the relevant parameters are not retrievable from the BSF according to the B-TID, or if the retrieval succeeds but the stored nonce and nc are inconsistent with the nonce and nc in the request message, the AuthS returns an error cause to the AS, requires the UE to perform a GBA process again, and re-negotiates the key data shared between the UE and the AS.

[0118] 5. The AS sends an error response message

"401 Unauthorized" to the UE. The message carries a WWW-Authenticate header, and the nonce value in the message is NULL, which instructs the UE to restart the GBA process.

[0119] 6. The UE receives the response message, and clears the shared key data negotiated in the previous GBA process.

[0120] 7. The GBA process described in section I above is performed between the UE and the BSF. After completion of the GBA process, the UE reinitializes the data required for authentication. The data is used to resend a request to the AS.

[0121] 8-13. Steps 8-13 are the same as steps 2-7 in "II. Authentication success process".

IV. Authentication failure process (2)

[0122] As shown in FIG. 12, the process includes the following steps:

[0123] 1-3. Steps 1-3 are the same as steps 1-3 in "II. Authentication success process".

[0124] 4. After the AuthS receives the request message, if the relevant data stored in the AuthS is retrieved unsuccessfully according to the B-TID and the nc is greater than 1, or if the relevant parameters are not retrievable from the BSF according to the B-TID, step 4 in section III described above serves as a reference. If the retrieval succeeds (including success of retrieving the AuthS storage and success of retrieving the BSF), and the Ks corresponding to the B-TID is stored but the stored nonce and nc are inconsistent with the nonce and nc in the request message, the nonce is reallocated and the value of the nc is initialized to 1.

[0125] 5. The AuthS returns a cause of error and the reallocated nonce to the AS.

[0126] 6. The AS sends an error response message "401 Unauthorized" to the UE. The message carries a WWW-Authenticate header, and the nonce in the message is the reallocated nonce of the AuthS, which instructs the UE to restart the process of requesting the AS.

[0127] 7. The UE receives the response message, and updates the nonce and nc.

[0128] 8-11. Steps 8-11 are the same as steps 9-12 in "II. Authentication success process".

[0129] It is understandable to those skilled in the art that all or part of the steps in the foregoing embodiments may be performed through hardware instructed by a program. The program may be stored in a computer-readable storage medium such as a Read-Only Memory/Random Access Memory (ROM/RAM), a magnetic disk, and a compact disk.

[0130] Detailed above are a method for an authentication service device to authenticate request messages, a method for an AS to authenticate requests of a UE, a UE authentication method, and the corresponding authentication service device, AS, and UE. Although the invention is described through some exemplary embodiments, the invention is not limited to such embodiments. It is apparent that those skilled in the art can make modifications and variations to the invention without departing from the spirit and scope of the invention. The invention is intended to cover the modifications and variations provided that they fall in the scope of protection defined by the following claims or their equivalents.

Claims

1. A method for authenticating request messages, comprising:

   negotiating key information with a User Equipment (UE), and allocating an authentication transaction identifier and an authentication random number corresponding to the key information to the UE;
   receiving an association request from an Application Server (AS), wherein the association request carries the authentication transaction identifier and the authentication random number;
   verifying the authentication random number carried in the association request according to the authentication transaction identifier carried in the association request; and
   searching for the key information corresponding to the authentication transaction identifier carried in the association request after the verification succeeds, and returning an association response to the AS in response to the association request, wherein the association response carries an authentication key generated according to the found key information.

2. The method of claim 1, wherein:

   the verifying of the authentication random number carried in the association request according to the authentication transaction identifier carried in the association request comprises:
   searching for the authentication random number corresponding to the authentication transaction identifier carried in the association request, and
   verifying whether the authentication random number carried in the association request is consistent with the found authentication random number; and
   the method further comprises: after the verification succeeds, allocating a new authentication random number to update the found authentication random number, wherein the association response further carries the new authentication random number which instructs the UE to use the new authentication random number to send an authentication request to a next AS.

3. The method of claim 1, wherein:

the association request further carries a use count of the authentication random number;
the verifying of the authentication random number carried in the association request according to the authentication transaction identifier carried in the association request comprises: searching for the authentication random number corresponding to the authentication transaction identifier carried in the association request and its use count, and verifying whether the authentication random number carried in the association request and its use count are consistent with the found authentication random number and its use count; and
the method further comprises: increasing the use count of the found authentication random number progressively after the verification succeeds.

4. The method of any of claims 1-3, further comprising:

returning an association failure response to the AS in response to the association request after the verification fails, wherein the association failure response carries an indication which indicates verification failure and instructs the UE to perform key negotiation again.

5. The method of any of claims 1-3, further comprising:

checking for existence of any key information corresponding to the authentication transaction identifier carried in the association request after the verification fails;
if such key information exists, reallocating an authentication random number as the authentication random number corresponding to the authentication transaction identifier carried in the association request, and returning an association reset response to the AS in response to the association request, wherein the association reset response carries the reallocated authentication random number which instructs the UE to use the reallocated authentication random number to resend the authentication request to the AS; or
if no such key information exists, returning an association failure response to the AS in response to the association request, wherein the association failure response carries an indication which indicates verification failure and instructs the UE to perform key negotiation again.

6. A method for an Application Server (AS) to authenticate a request of a User Equipment (UE), comprising:

receiving an authentication request from the UE, wherein the authentication request carries an authentication transaction identifier and an authentication random number;
sending an association request to an authentication service device according to the authentication request, wherein the association request carries the authentication transaction identifier and the authentication random number;
receiving an association response returned by the authentication service device in response to the association request, wherein the association response carries an authentication key; and
returning a success response to the UE in response to the authentication request, wherein the success response carries authentication response data generated according to the authentication key.

7. The method of claim 6, wherein:

the association response further carries a new authentication random number; and
the success response returned to the UE in response to the authentication request further carries the new authentication random number for instructing the UE to use the new authentication random number to send an authentication request to a next AS.

8. The method of claim 6, wherein:

the authentication request further carries a use count of the authentication random number; and
the association request sent to the authentication service device according to the authentication request further carries the use count of the authentication random number.

9. The method of any of claims 6-8, wherein: the authentication request further carries a domain name of a server which authenticates the request; and the method further comprises:

checking whether the domain name of the server which authenticates the request is a domain name of the authentication service device before sending the association request to the authentication service device according to the authentication request; and
sending the association request to the authentication service device if the domain name of the server which authenticates the request is the domain name of the authentication service device.

10. The method of any of claims 6-8, further comprising:

receiving an association failure response re-

turned by the authentication service device in response to the association request, and returning a failure response to the UE in response to the authentication request according to the association failure response, wherein the failure response carries an indication which requires the UE to perform key negotiation again; or
receiving an association reset response returned by the authentication service device in response to the association request, wherein the association reset response carries a reallocated authentication random number corresponding to the authentication transaction identifier carried in the association request; returning a reset response to the UE in response to the authentication request, wherein the reset response carries the reallocated authentication random number for instructing the UE to use the reallocated authentication random number to resend the authentication request.

11. A User Equipment (UE) authentication method, comprising:

sending an authentication request to an Application Server (AS), wherein the authentication request carries an authentication transaction identifier and an authentication random number corresponding to key information;
receiving a success response returned by the AS in response to the authentication request, wherein the success response carries authentication response data; and verifying the authentication response data according to the key information.

12. The UE authentication method of claim 11, further comprising:

obtaining the authentication transaction identifier and the authentication random number corresponding to the key information from locally stored information.

13. The UE authentication method of claim 11, further comprising:

negotiating the key information with an authentication service device and obtaining the authentication transaction identifier and the authentication random number corresponding to the key information.

14. The UE authentication method of claim 11, wherein:

the success response further carries a new authentication random number; and
the method further comprises: using the new authentication random number to update the stored authentication random number.

15. The UE authentication method of claim 11, wherein:

the authentication request further carries a use count of the authentication random number;
the method further comprises: increasing the use count of the stored authentication random number progressively after receiving the success response.

16. The UE authentication method of claim 11, further comprising:

receiving a failure response returned by the AS in response to the authentication request, wherein the failure response carries an indication which instructs the UE to perform key negotiation again; and
clearing the stored key information and the corresponding authentication transaction identifier and authentication random number according to the indication, re-negotiating the key information with the authentication service device, and then using the re-negotiated key information to send the authentication request to the AS.

17. The UE authentication method of claim 11, further comprising:

receiving a reset response returned by the AS in response to the authentication request, wherein the reset response carries a reallocated authentication random number; using the reallocated authentication random number to update the stored authentication random number, and then using the reallocated authentication random number to send the authentication request to the AS.

18. An authentication service device, comprising:

an authentication service module, adapted to: negotiate key information with a User Equipment (UE), and allocate an authentication transaction identifier and an authentication random number corresponding to the key information to the UE; and
a uniform authentication module, adapted to: receive an association request from an Application Server (AS), and obtain the authentication transaction identifier and the authentication random number carried in the association request; verify the authentication random number carried in the association request according to the authentication transaction identifier carried in the association request and data provided by the authen-

tication service module; and search for the key information corresponding to the authentication transaction identifier carried in the association request after the verification succeeds, and return an association response to the AS in response to the association request, wherein the association response carries an authentication key generated according to the found key information.

19. The authentication service device of claim 18, wherein:

the uniform authentication module is further adapted to: store the data found in the authentication service module for verifying the authentication random number carried in the association request; and
when the uniform authentication module verifies the authentication random number carried in the association request, the uniform authentication module queries locally stored data first, and, if the query fails, queries the authentication service module.

20. The authentication service device of claim 19, wherein:

the verification performed by the uniform authentication module for the authentication random number carried in the association request verifies whether the authentication random number carried in the association request is consistent with the stored authentication random number; and
the uniform authentication module is further adapted to: allocate a new authentication random number to update the stored authentication random number after the verification succeeds, and add the new authentication random number into the association response for instructing the UE to use the new authentication random number to send an authentication request to a next AS.

21. The authentication service device of claim 19, wherein:

the uniform authentication module is further adapted to obtain a use count of the authentication random number carried in the association request;
the verification performed by the uniform authentication module for the authentication random number carried in the association request verifies whether the authentication random number carried in the association request and its use count are consistent with the stored authentication random number and its use count; and
the uniform authentication module is further adapted to increase the stored use count of the authentication random number progressively after the verification succeeds.

22. The authentication service device of claim 19, wherein:

the uniform authentication module is further adapted to: return an association failure response to the AS in response to the association request after the verification fails, wherein the association failure response carries an indication which indicates verification failure and instructs the UE to perform key negotiation again; or
the uniform authentication module is further adapted to: check for existence of any key information corresponding to the authentication transaction identifier carried in the association request after the verification fails; if such key information exists, reallocate an authentication random number to update the stored authentication random number corresponding to the authentication transaction identifier carried in the association request, and return an association reset response to the AS in response to the association request, wherein the association reset response carries the reallocated authentication random number which instructs the UE to use the reallocated authentication random number to resend the authentication request to the AS; or, if no such key information exists, return an association failure response to the AS in response to the association request.

23. An Application Server (AS), comprising:

an interface module, adapted to: receive an authentication request from a User Equipment (UE); send an association request to an authentication service device; receive an association response returned by the authentication service device in response to the association request; and return a success response to the UE in response to the authentication request; and
a security association module, adapted to: obtain an authentication transaction identifier and an authentication random number carried in the authentication request, generate the association request according to the authentication request, wherein the association request needs to be sent to the authentication service device and carries the authentication transaction identifier and the authentication random number; obtain an authentication key carried in the association

response, generate authentication response data according to the authentication key carried in the association response, and generate the success response that needs to be returned to the UE, wherein the success response carries the authentication response data.

24. The AS of claim 23, wherein:

the security association module is further adapted to: obtain a domain name of a server which authenticates the authentication request, wherein the domain name is carried in the authentication request; check that the domain name of the server which authenticates the authentication request is the domain name of the authentication service device, and then generate the association request that needs to be sent to the authentication service device.

25. The AS of claim 23, wherein:

the interface module is further adapted to: receive an association failure response or an association reset response returned by the authentication service device in response to the association request; and return a failure response or a reset response to the UE in response to the authentication request; and
the security association module is further adapted to: generate the failure response that needs to be returned to the UE according to the association failure response, wherein the failure response carries an indication which instructs the UE to perform key negotiation again; obtain a reallocated authentication random number carried in the association reset response; and generate the reset response that needs to be returned to the UE according to the association reset response, wherein the reset response carries the reallocated authentication random number for instructing the UE to use the reallocated authentication random number to resend the authentication request.

26. A User Equipment (UE), comprising:

a security service module, adapted to: negotiate key information with an authentication service device, and obtain an authentication transaction identifier and an authentication random number which are allocated by the authentication service device and correspond to the key information;
an application service module, adapted to: send an authentication request to an Application Server (AS), wherein the authentication request carries the authentication transaction identifier and the authentication random number; and obtain a success response returned by the AS in response to the authentication request, obtain authentication response data carried in the success response, and verify the authentication response data according to the key information; and
an authentication control module, adapted to: judge whether the key information and the authentication transaction identifier and the authentication random number corresponding to the key information are stored when an authentication request needs to be sent to the AS; if no such information is stored, control the security service module to obtain required data; and, if such information is stored, control the application service module to send the authentication request.

27. The UE of claim 26, wherein:

the application service module is further adapted to obtain a new authentication random number from the success response; and
the authentication control module is further adapted to use the new authentication random number to update the stored authentication random number.

28. The UE of claim 26, wherein:

the application service module is further adapted to add a use count of the authentication random number into the authentication request; and
the authentication control module is further adapted to increase the use count of the stored authentication random number progressively after receiving the success response.

29. The UE of claim 26, wherein:

the application service module is further adapted to: obtain a failure response returned by the AS in response to the authentication request, wherein the failure response carries an indication which instructs the UE to perform key negotiation again; and
the authentication control module is further adapted to: clear the stored key information and the corresponding authentication transaction identifier and authentication random number according to the indication in the failure response, control the security service module to obtain the required data again, and then control the application service module to resend the authentication request.

30. The UE of claim 26, wherein:

the application service module is further adapted to: obtain a reset response returned by the AS in response to the authentication request, and obtain a reallocated authentication random number carried in the reset response; and
the authentication control module is further adapted to: use the reallocated authentication random number to update the stored authentication random number, and then control the application service module to resend the authentication request.

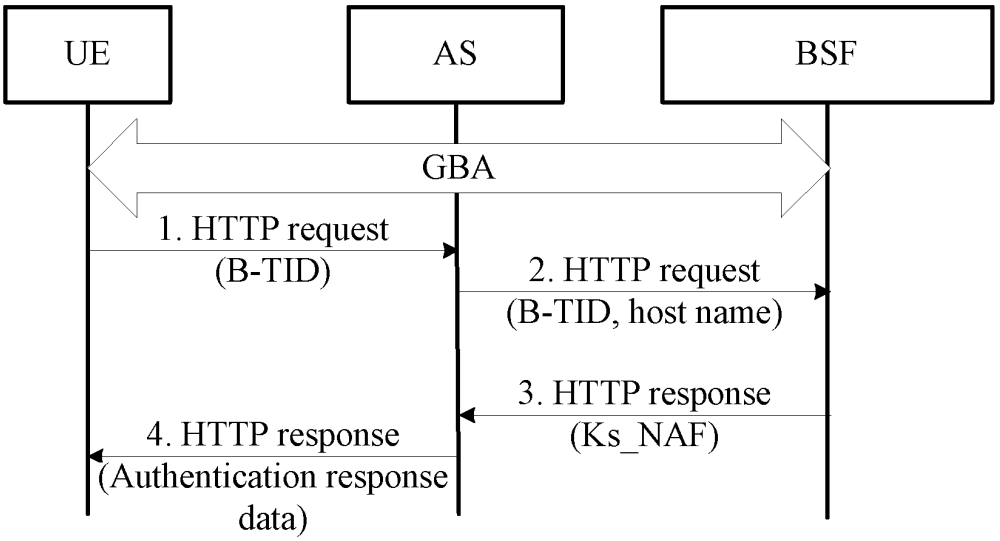
UE
AS
BSF
GBA
1. HTTP request
(B-TID)
2. HTTP request
(B-TID, host name)
3. HTTP response
(Ks_NAF)
4. HTTP response
(Authentication response
data)

FIG. 1

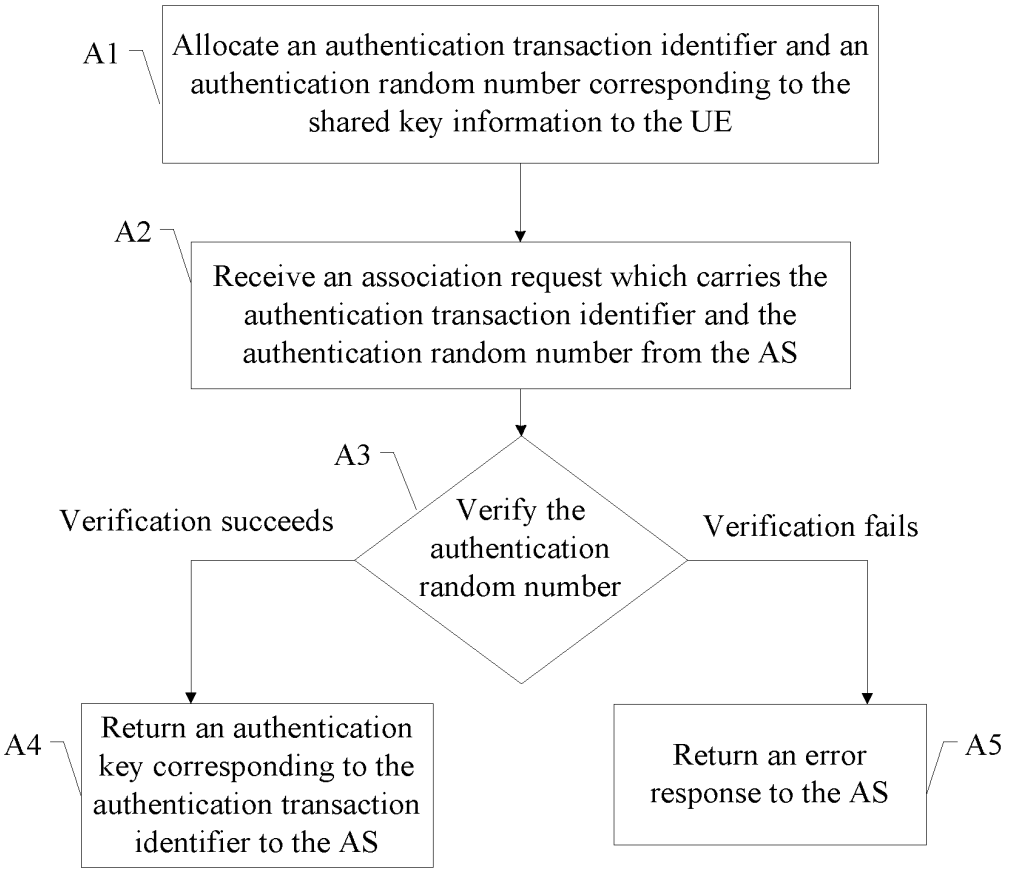
A1
Allocate an authentication transaction identifier and an authentication random number corresponding to the shared key information to the UE
A2
Receive an association request which carries the authentication transaction identifier and the authentication random number from the AS
A3
Verify the authentication random number
Verification succeeds
Verification fails
A4
Return an authentication key corresponding to the authentication transaction identifier to the AS
Return an error response to the AS
A5

FIG. 2

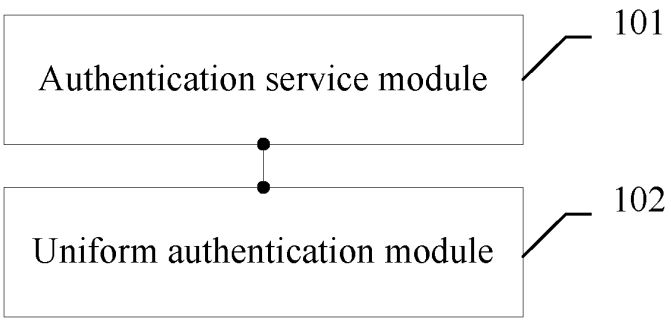
Authentication service module
101
Uniform authentication module
102

FIG. 3

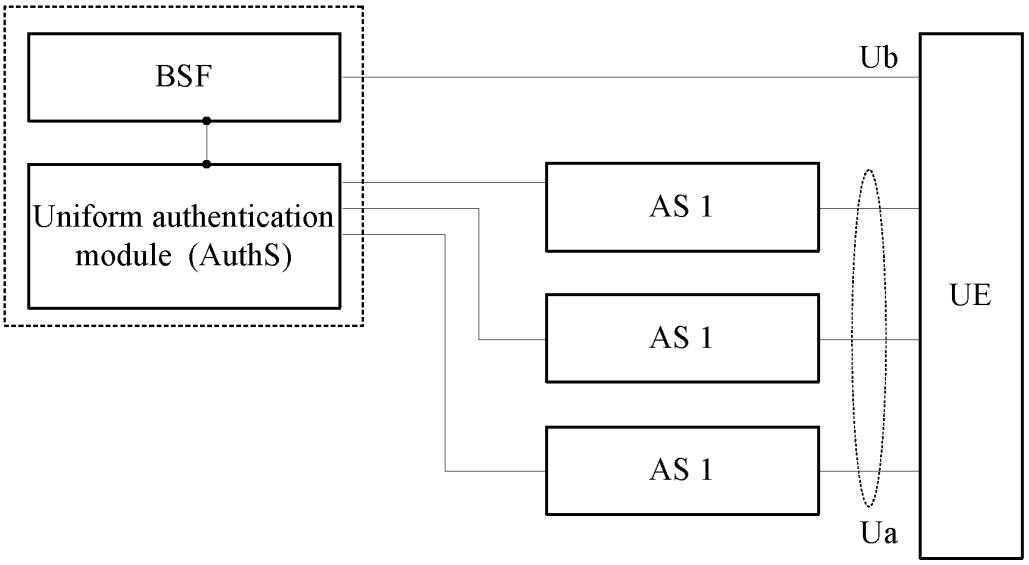
BSF
Uniform authentication module (AuthS)
Ub
AS 1
AS 1
AS 1
UE
Ua

FIG. 4

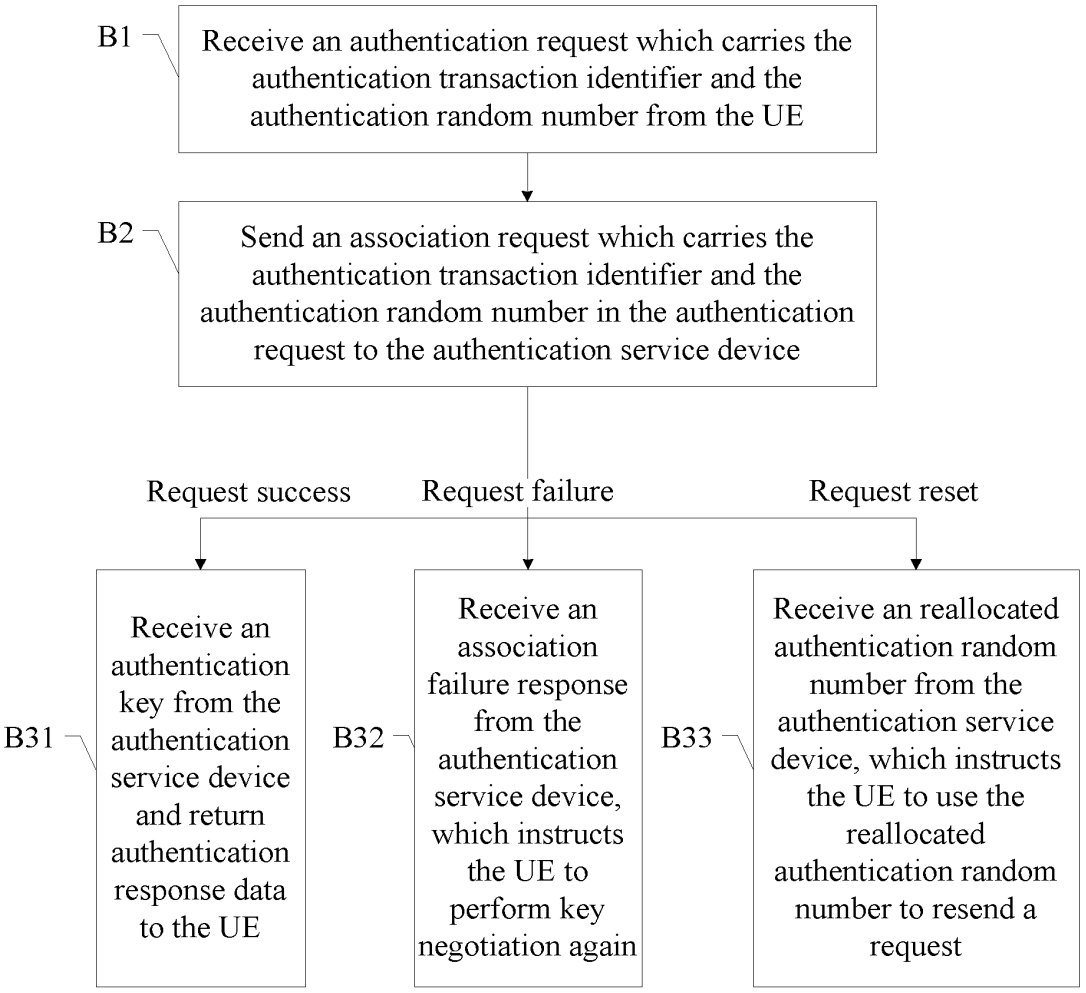
B1
Receive an authentication request which carries the authentication transaction identifier and the authentication random number from the UE
B2
Send an association request which carries the authentication transaction identifier and the authentication random number in the authentication request to the authentication service device
Request success
Request failure
Request reset
B31
Receive an authentication key from the authentication service device and return authentication response data to the UE
B32
Receive an association failure response from the authentication service device, which instructs the UE to perform key negotiation again
B33
Receive an reallocated authentication random number from the authentication service device, which instructs the UE to use the reallocated authentication random number to resend a request

FIG. 5

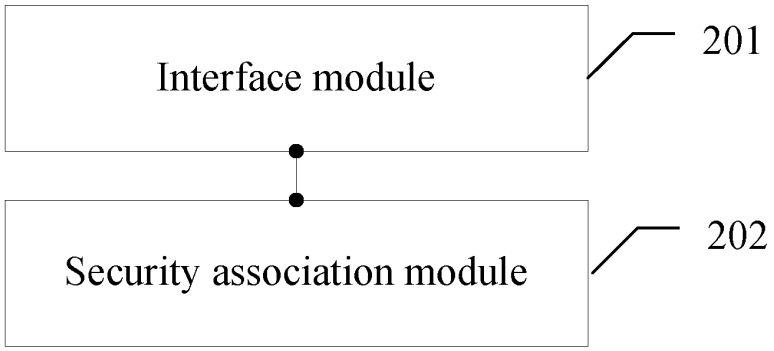
Interface module
201
Security association module
202

FIG. 6

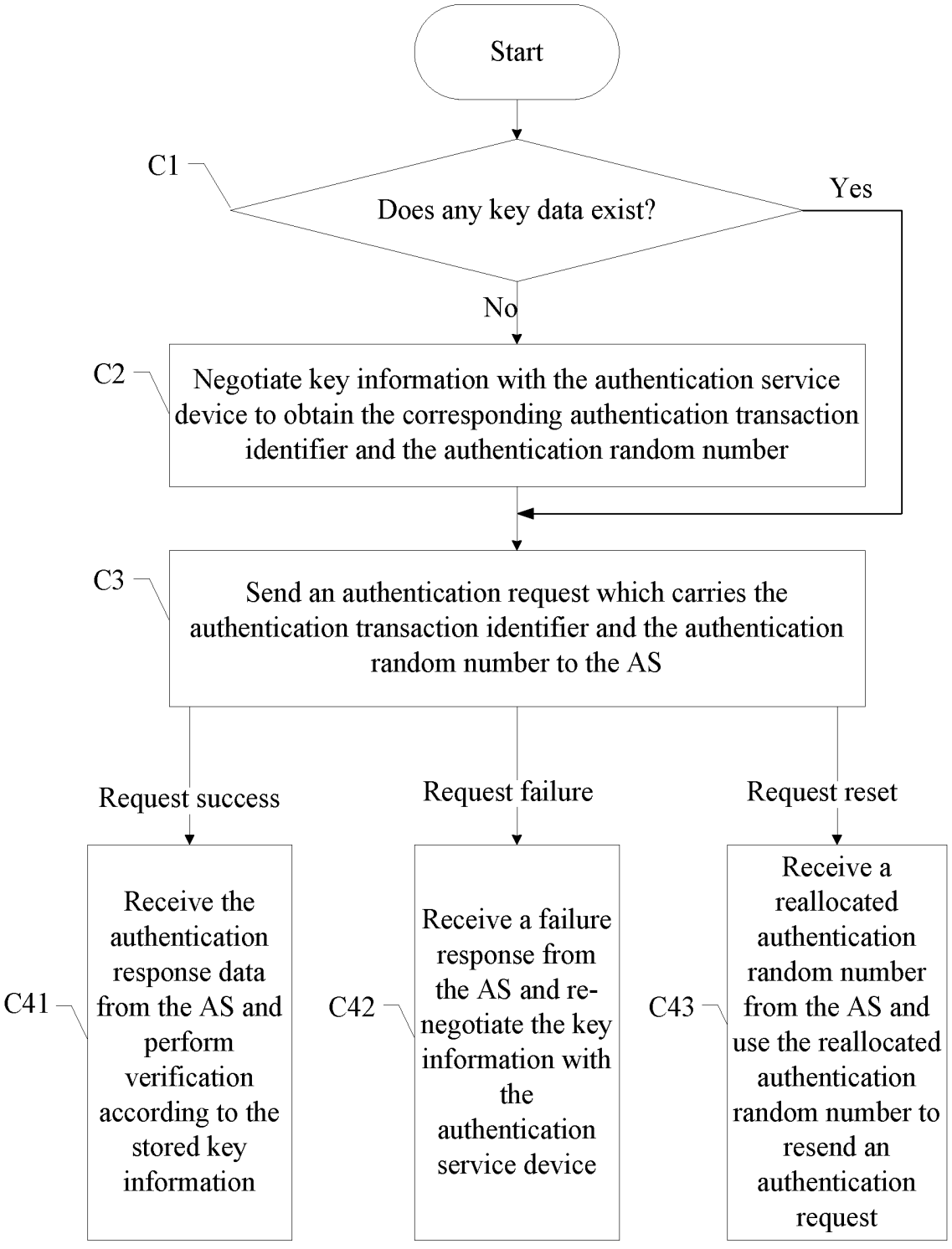
Start
C1
Does any key data exist?
Yes
No
C2
Negotiate key information with the authentication service device to obtain the corresponding authentication transaction identifier and the authentication random number
C3
Send an authentication request which carries the authentication transaction identifier and the authentication random number to the AS
Request success
Request failure
Request reset
C41
Receive the authentication response data from the AS and perform verification according to the stored key information
C42
Receive a failure response from the AS and re-negotiate the key information with the authentication service device
C43
Receive a reallocated authentication random number from the AS and use the reallocated authentication random number to resend an authentication request

FIG. 7

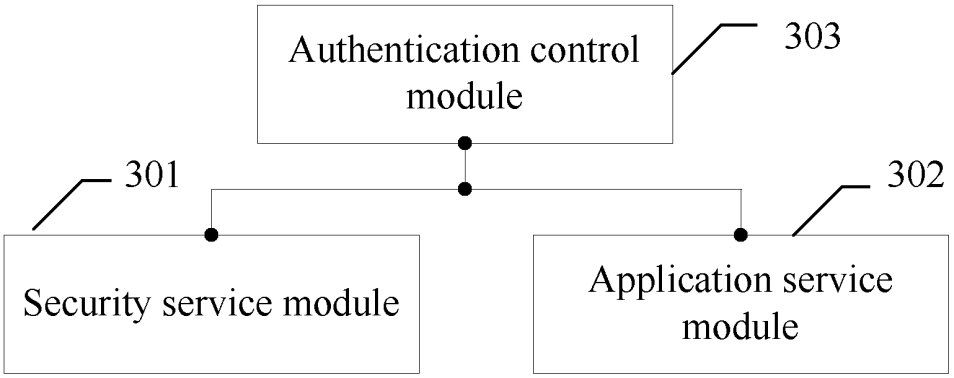
Authentication control module
303
301
302
Security service module
Application service module

FIG. 8

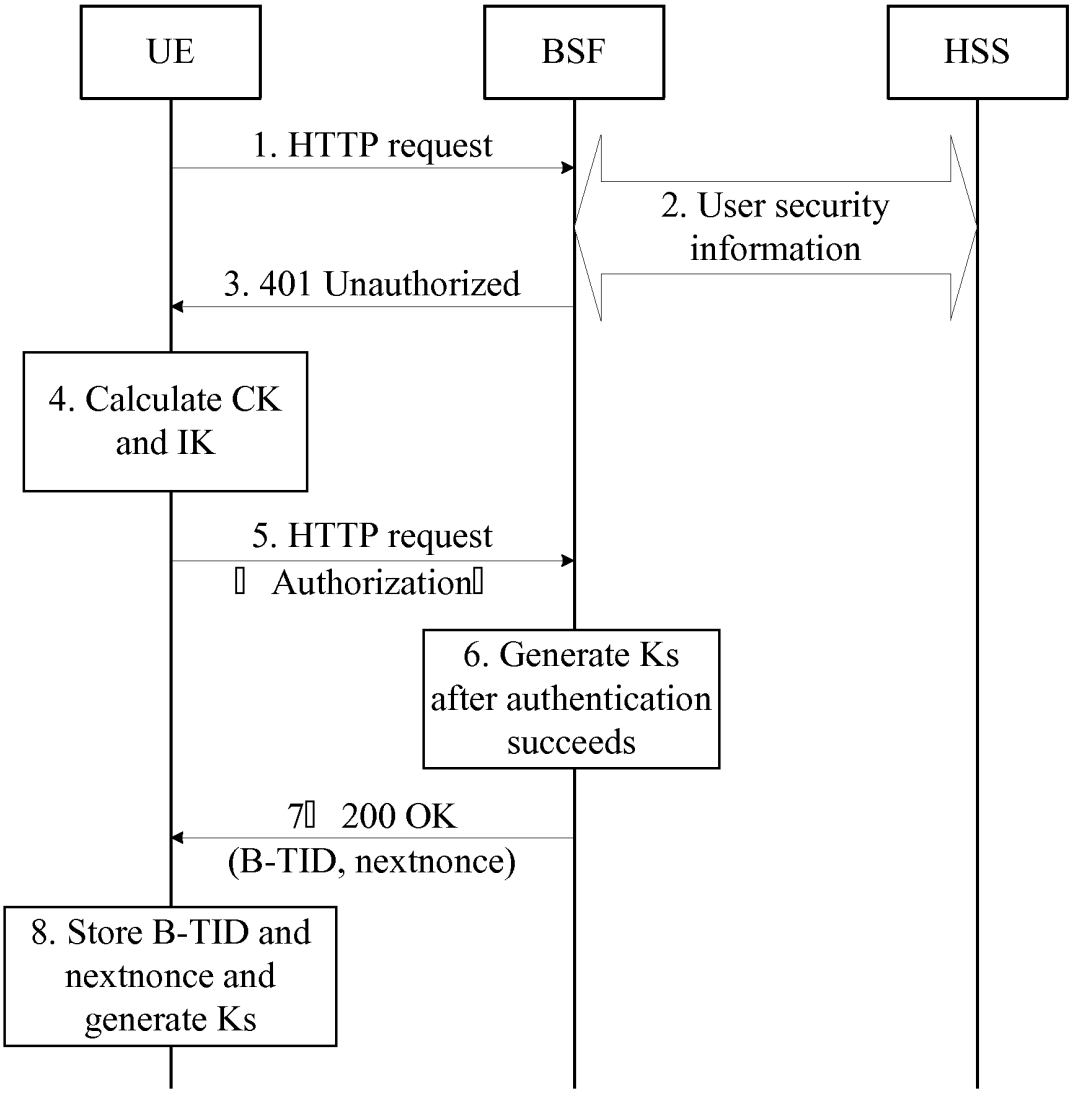
UE
BSF
HSS
1. HTTP request
2. User security information
3. 401 Unauthorized
4. Calculate CK and IK
5. HTTP request
Authorization
6. Generate Ks after authentication succeeds
7 200 OK
(B-TID, nextnonce)
8. Store B-TID and nextnonce and generate Ks

FIG. 9

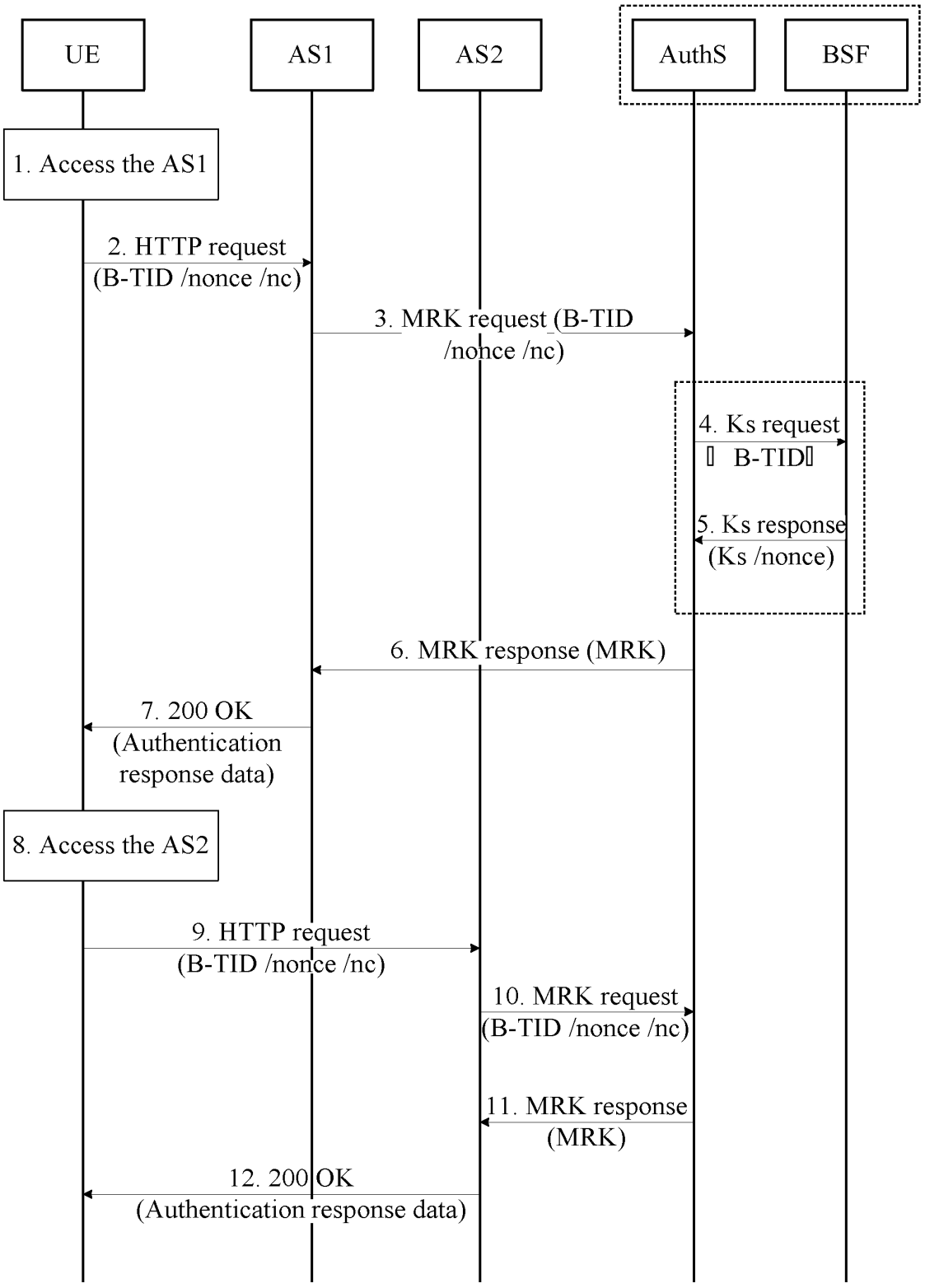
UE
AS1
AS2
AuthS
BSF
1. Access the AS1
2. HTTP request
(B-TID /nonce /nc)
3. MRK request (B-TID
/nonce /nc)
4. Ks request
B-TID
5. Ks response
(Ks /nonce)
6. MRK response (MRK)
7. 200 OK
(Authentication
response data)
8. Access the AS2
9. HTTP request
(B-TID /nonce /nc)
10. MRK request
(B-TID /nonce /nc)
11. MRK response
(MRK)
12. 200 OK
(Authentication response data)

FIG. 10

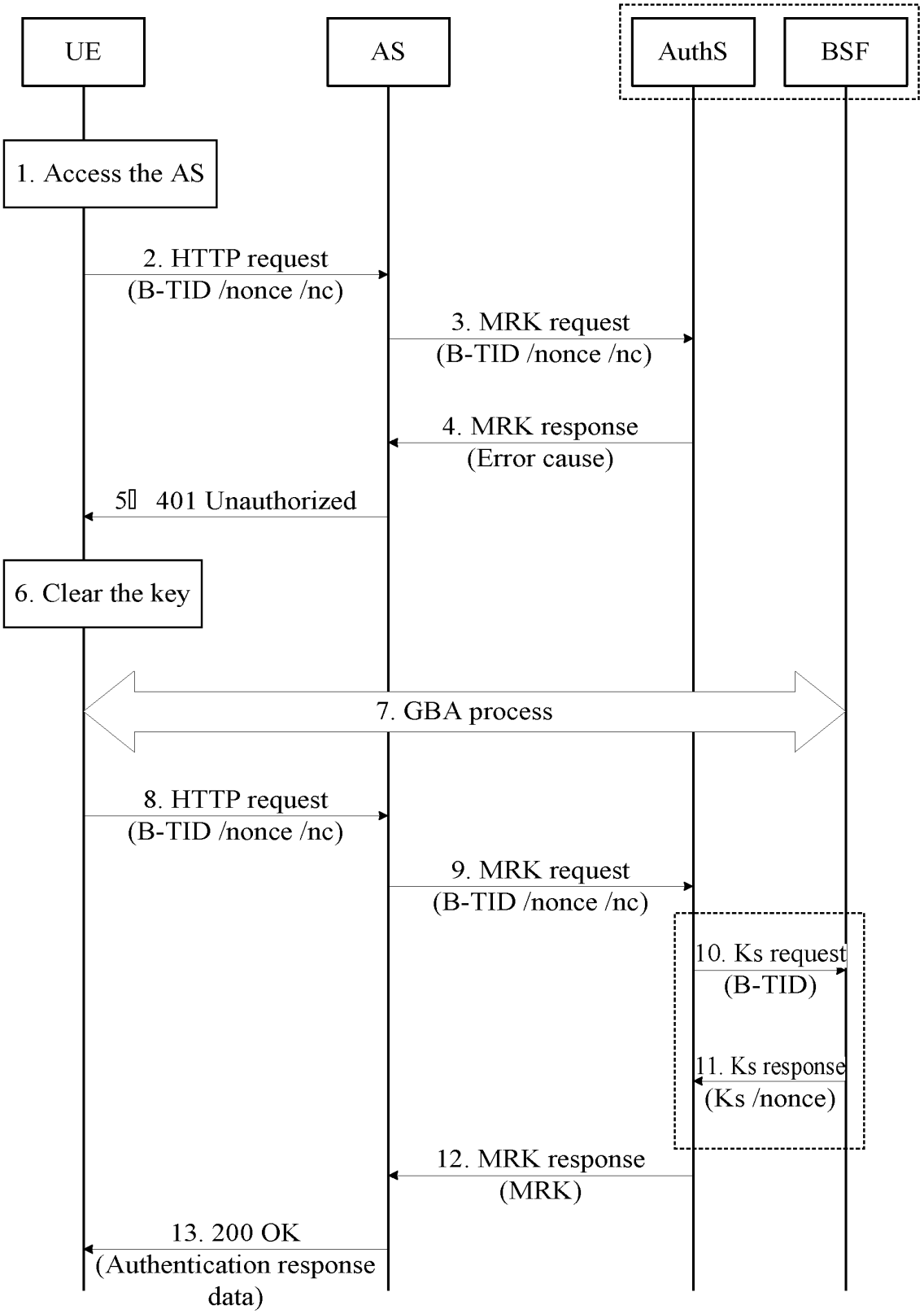
UE
AS
AuthS
BSF
1. Access the AS
2. HTTP request
(B-TID /nonce /nc)
3. MRK request
(B-TID /nonce /nc)
4. MRK response
(Error cause)
5. 401 Unauthorized
6. Clear the key
7. GBA process
8. HTTP request
(B-TID /nonce /nc)
9. MRK request
(B-TID /nonce /nc)
10. Ks request
(B-TID)
11. Ks response
(Ks /nonce)
12. MRK response
(MRK)
13. 200 OK
(Authentication response data)

FIG. 11

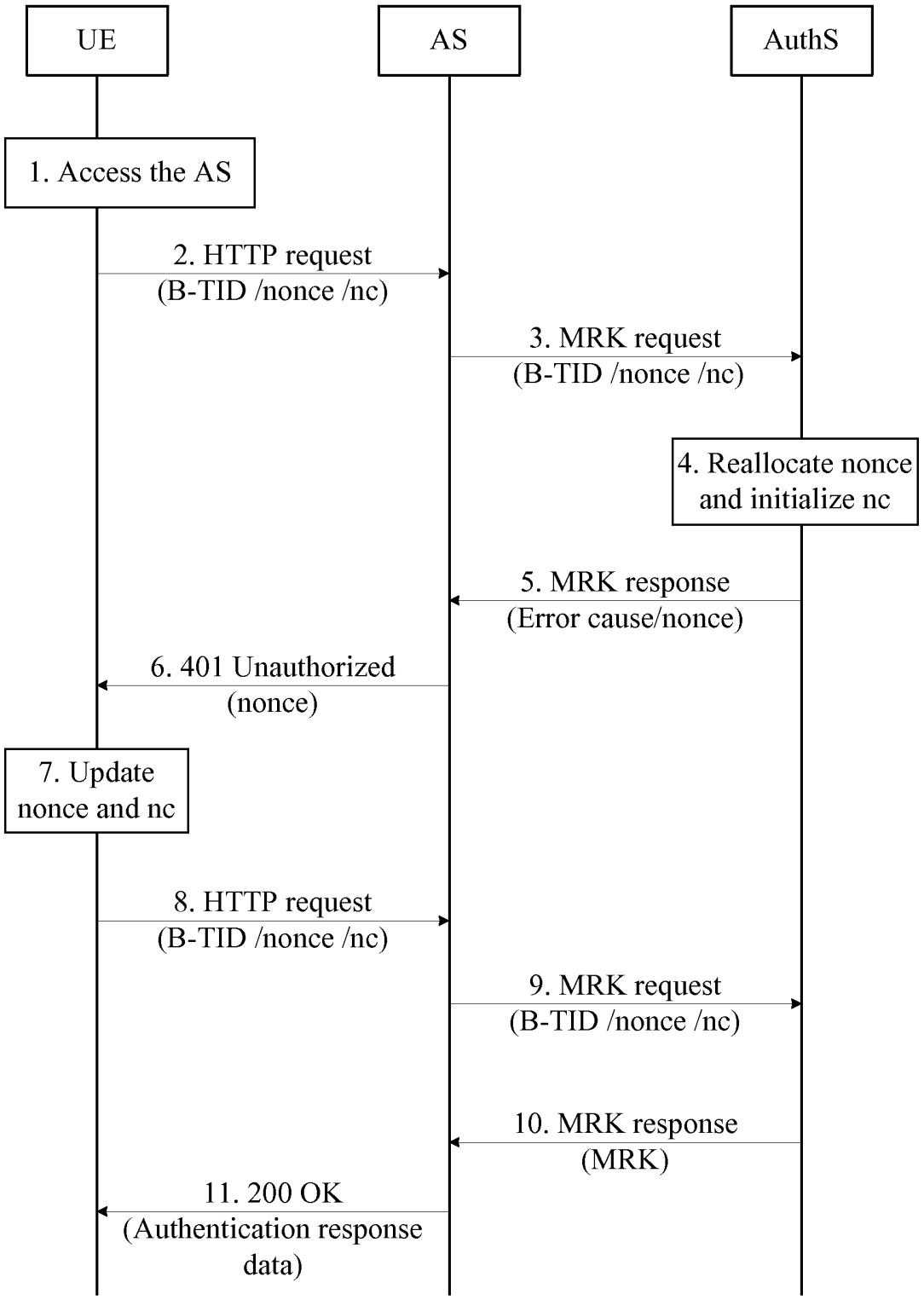
UE
AS
AuthS
1. Access the AS
2. HTTP request
(B-TID /nonce /nc)
3. MRK request
(B-TID /nonce /nc)
4. Reallocate nonce and initialize nc
5. MRK response
(Error cause/nonce)
6. 401 Unauthorized
(nonce)
7. Update nonce and nc
8. HTTP request
(B-TID /nonce /nc)
9. MRK request
(B-TID /nonce /nc)
10. MRK response
(MRK)
11. 200 OK
(Authentication response data)

FIG. 12

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2008/071894

A. CLASSIFICATION OF SUBJECT MATTER

H04L 9/00 (2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: H04L 9, H04L 29, H04M 11

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CPRS, CNKI, WPI, EPODOC, PAJ: identifier, id, random w number?, authentica+, authori+, key, request, IMS

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
P, X	CN101163010A (HUAWEI TECHNOLOGIES CO LTD) 16 Apr. 2008 (16.04.2008) the whole document	1-30
X	CN1801697A (HUAWEI TECHNOLOGIES CO LTD) 12 Jul. 2006 (12.07.2006) description page 3 line 24 to page 4 line 12, figs. 4-5	6,11,23,26
A		1-5,12-22,24-25,27-30
A	CN1845600A (CHINA MOBILE COMMUNICATIONS CORP) 11 Oct. 2006 (11.10.2006) the whole document	1-30
A	CN1756428A (HUAWEI TECHNOLOGIES CO LTD) 05 Apr. 2006 (05.04.2006) the whole document	1-30
A	WO2007026914A1 (TELEFONAKTIEBOLAGET ERICSSON L M) 08 Mar. 2007 (08.03.2007) the whole document	1-30

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim (S) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
03 Nov. 2008 (03.11.2008)	**20 Nov. 2008 (20.11.2008)**
Name and mailing address of the ISA/CN The State Intellectual Property Office, the P.R.China 6 Xitucheng Rd., Jimen Bridge, Haidian District, Beijing, China 100088 Facsimile No. 86-10-62019451	Authorized officer HAO Aixin Telephone No. (86-10)62411443

Form PCT/ISA/210 (second sheet) (April 2007)

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2008/071894

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN101163010A	16.04.2008	NONE	
CN1801697A	12.07.2006	WO2006072209A1	13.07.2006
CN1845600A	11.10.2006	NONE	
CN1756428A	05.04.2006	CN100384120C	23.04.2008
WO2007026914A1	08.03.2007	EP1920392A	14.05.2008

Form PCT/ISA/210 (patent family annex) (April 2007)

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- CN 200710165986 **[0001]**